VM Service

A Benchmark suite for cloud environment

USER'S MANUAL

May 27th 2013

Revision Sheet

Release No.	**Date**	**Revision Description**
Rev. 1.0	27/05/2013	VM Service v1.0

Content

1 Introduction

The VM Service benchmark suite includes two services, which have already deployed the service end on the Virtual Machine. The two services are search engine and media streaming. We provide the Virtual Machine images (domain U), so that the users can download them and deploy the images directly on their Virtual Machine Monitor (VMM).

2 Virtual Environment

For the current version, we provide the XCP format virtual machine images (Domain U). The Xen Cloud Platform (or XCP) is a turnkey open source virtualization solution that provides out-of-the box virtualization and cloud computing. XCP includes the Xen Hypervisor, the enterprise ready Xen API toolstack and integrations for cloud, storage and networking solutions[1]. For more information about XCP, users can find from here.

2.1 VM image

We provide three VM images for the user to download and import to their own virtual environments. The Table 1 lists the VM image name and corresponding services it provide.

Table 1 VM image information

VM image name	Description
SearchFront.xva	Search benchmark front end service
SearchBack.xva	Search benchmark back end service
MediaStreaming.xva	Media Streaming service

2.2 Import the VM

The users need to import the VM images to their own XCP environments by using the

[1] XCP main page http://www.xen.org/products/cloudxen.html

following command:

xe vm-import filename=/path/to/your/image

2.3 Prerequisites After importing the VM images

After importing the images, there are some configuration files user needed to modify. Users should pay attention that the images we provided are only the services. Users should deploy the corresponding workload generator to drive the services. How to deploy the workload generator will not be discussed in this guide, for we have detailed them in the other two user manual (**Search** and **Streaming**).

2.3.1 Setup passphraseless ssh

SSH must be installed and sshd must be running. To run the workload generator scripts that manage remote daemons, please make sure that you can ssh on VM (Domain U) without entering password.

If you cannot ssh to VMs without a passphrase, execute the following commands at workload generator node

$ ssh-keygen -t dsa -f $HOME/.ssh/id_dsa -P ''
This should result in two files, $HOME/.ssh/id_dsa (private key) and
$HOME/.ssh/id_dsa.pub (public key).
Copy $HOME/.ssh/id_dsa.pub to **service VMs (Domain U)**
On those nodes run the following commands:
$ cat id_dsa.pub >> $HOME/.ssh/authorized_keys2
$ chmod 0600 $HOME/.ssh/authorized_keys2
Depending on the version of OpenSSH the following commands may also be required:
$ cat id_dsa.pub >> $HOME/.ssh/authorized_keys
$ chmod 0600 $HOME/.ssh/authorized_keys
An alternative is to create a link from authorized_keys2 to authorized_keys:
$ cd $HOME/.ssh && ln -s authorized_keys2 authorized_keys
On the **workloads generator** node test the results by ssh'ing to other nodes:
$ ssh -i $HOME/.ssh/id_dsa server

2.3.2 Network and Host Name

This should come as no surprise, but for the sake of completeness we have to point out that all the machines must be able to reach each other over the network. User should modify the following files according to the experiment environments.

Modify the VM's IP address and Hostname:

Users should assign an IP address to each VM by modifying the file
/etc/sysconfig/network-scripts/ifcfg-eth0

```
#/etc/sysconfig/network-scripts/ifcfg-eth0
DEVICE=eth0
BOOTPROTO=static
IPADDR=172.18.12.204 # modify this line according to your environment
NETMASK=255.255.0.0 #modify the netmask
ONBOOT=yes
```

Users may need to change hostname of the VMs by modifying the file */etc/sysconfig.*

```
#/etc/sysconfig
NETWORKING=yes
NETWORKING_IPV6=no
HOSTNAME=vm004 #modify the hostname
```

Users should write the IP address and the corresponding hostname into */etc/hosts* for the entire node using in the experiment environment. The following is an example.

```
#/etc/hosts
127.0.0.1 localhost localhost.localdomain
172.18.11.75    Dom0
172.18.12.201   vm001
172.18.12.202   vm002
172.18.12.203   vm003
172.18.12.204   vm004
172.18.12.205   vm005
172.18.12.206   vm006
```

3 Run Search

The Search service is consisted with two parts, a web server (front end), which receives HTTP requests from clients and dispatching them to search servers, and a search server (back end), which serves client requests transmitting by Web Server and the return the results to web server. Users need import the ***SearchFront.xva*** and ***SearchBack.xva*** to users' XCP environment and deploy the corresponding workload generator.

3.1 Configuration

The user should deploy the workload generator before running the test. For how to deploy the workload generator users can consult to **Search manual**.

Here we specify some details the user should modify in workloads generator node.

(1) Modifying the file */Path/to/Search_benchmark/common.mk*。

Modifying line 17 and line1：

a) Line 17 web server's hostname

Master=vm004cp001 #web server (front end) 's hostname

b) Line 18 SearchServer (back end)'s hostname

Node=vm005cp001

(2) Entering Search directory */path/to/Search_benchmark/* and enter the following command

make cfg；

(3) Entering Search directory */path/to/Search_benchmark/*, the following file to search server's corresponding user's *nutch/* directory；

scp common.mk search_server :nutch/
scp nutch/si2.cfg search_server :nutch/

(4) Modifying workload generator 's file */path/to/Search_benchmark/exp/run-test.sh* line 28：

report search.throughputreal.head:1000000-fixed:100@s1i2@reqs-SoGou
for how to modify this line, please consult Search manual

3.2 Start test

Entering the directory */parth/to/Search_benchmark/*, and execute maketest to run the hole benchmark.

If the benchmark is running correctly, you can find a nutch service in the web site (http://YourWebWerveIP:9090)

4 Run media streaming

Users need to import the ***MediaStream.xva*** to users' XCP environment and deploy the corresponding workload generator.

4.1 Configuration

The user should deploy the workload generator before running the test. For how to deploy the workload generator users can consult to Media Streaming manual.

Users should modify the file in workload generator node. Modifying the file */path/to/faban/directory/streaming/deploy/run.xml* ：

(1) Modifying line 30，500 means the concurrent users' number

<fa:scale>500</fa:scale>

(2) Modifying line 34, 100000 means stable running time in second

<fa:steadyState>100000</fa:steadyState>

(3) Modifying line 162，put he media streaming VM's IP here, like 172.18.12.151

<ipAddress>172.18.12.151</ipAddress>

4.2 Start test

On the MediaStreaming service VM, user should start the ***Darwin*** service first by executing the following command.

```
/usr/local/sbin/DarwinStreamingServer -dDS 1
```

If there is already a service on the VM, users can use the following command to kill it.

```
killall /usr/local/sbin/DarwinStreamingServer
```

And then the users can start the test by executing the following command on workload generator node.

```
/path/to/faban/directory/streaming/scripts/run-test.sh
```